Solartron User Function

What is the Solartron user function?

Technical details.

What are the advantages and disadvantages?

User function safeguards.

What has been accomplished?

Sample output data.

Future ideas.

What is the Solartron user function?

The User Function facility of the 1051 enables functionality to be added to the system which is not available on the standard 1051, providing significant potential for customizing the data manipulation capabilities of 1051 to suit the requirements of the user.

Typical uses of User Functions could be to:

* Calculate a special parameter for which there is no standard implementation on 1051;

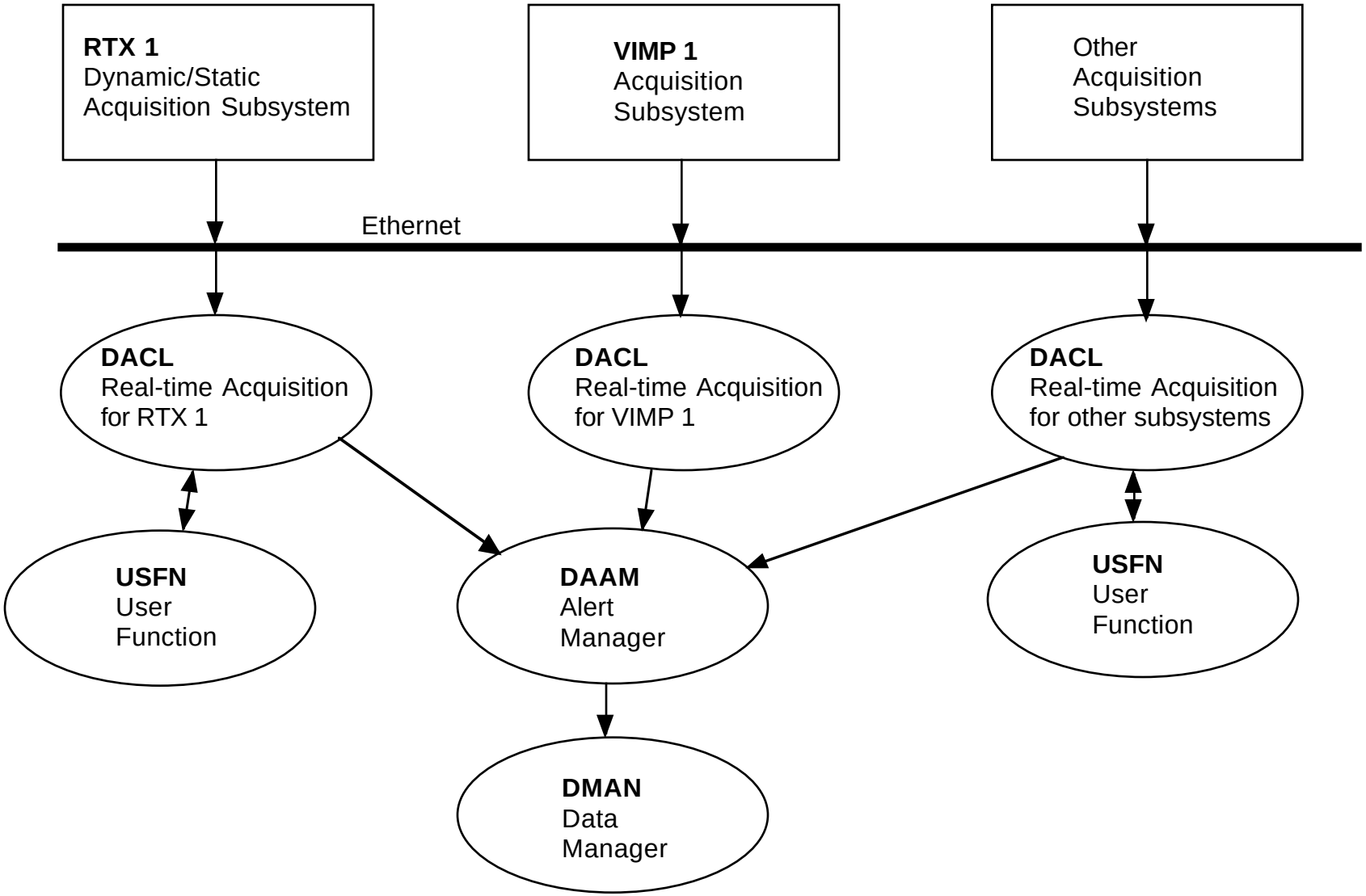
* Calculate variable alarm limits based on a static parameter. For example, the limits on vibration channels could be widened as the temperature increases.

How does the user function operate?

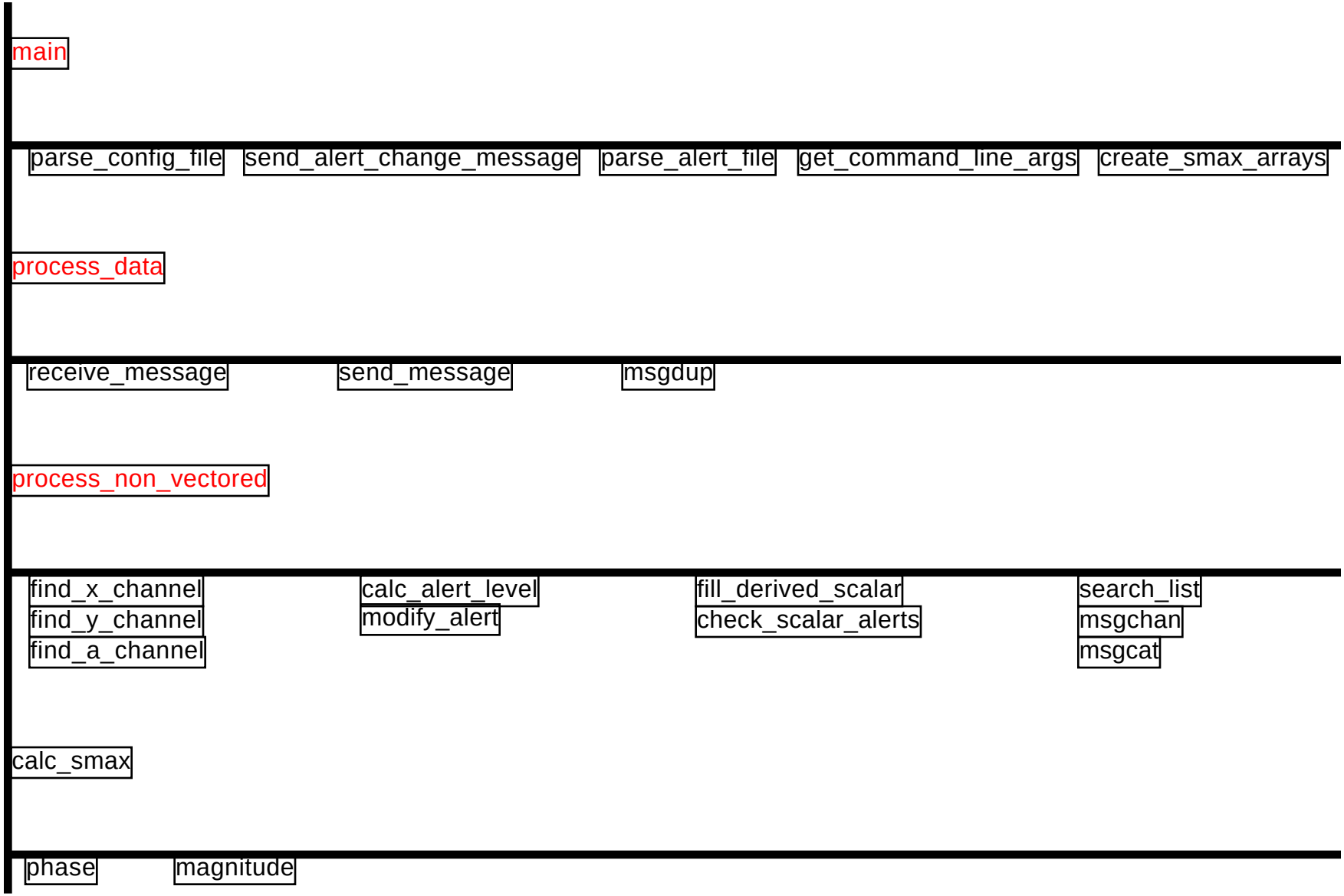
User functions (USFNs) effectively "intercept" data between acquisition and the database. When no USFNs are in operation, data are collected by the 1051 real-time acquisition module (known as DACL) from an acquisition subsystem and from the Data Import mechanism. DACL checks to see if any alerts have occurred, then the data are passed to the Alert Manager (DAAM) to initiate any actions arising from alerts, and then to the real-time storage module (known as DMAN). There is one instance of DACL for each acquisition subsystem.

A USFN is made active when DACL detects that the configuration file user_ch is present. In this case, data from DACL are redirected to the USFN module, where they are processed and then returned back to DACL. From here data are sent to DAAM and DMAN.

User Function Process Diagram



User Function Calling Diagram



User Function Advantages

The user function is capable of performing any task permitted by the Solartron 1051 product software, the C programming language, and the Unix operating system, subject to some limitations, such as timing constraints. All of the features and internal data of the Solartron 1051 software are available, as are the commands and facilities of the unix operating system.

User Function Disadvantages

Since the USFN code runs on the live machine and operates in the main data path, it is possible for coding errors in the module to have a severe impact on the 1051 system. Incorrect coding may also result in invalid data being stored in the database. Improper operation of the USFN may cause malfunction or collapse of the 1051 software, loss of data, or storage of erroneous data.

User function design and implementation is not a trivial process, as it involves writing or modifying C programs, creating a new configuration file, editing other configuration files, and rebuilding the 1051 software. It should only be undertaken if one has a good knowledge and understanding of UNIX and C. Great care must be taken to follow design rules in order not to corrupt the standard 1051 data structures: it is not a process to be lightly undertaken.

User Function Safeguards

1. Returned user data does not conform to expectations.
Result: USFN data are discarded, and original data are retained.

2. Detection of system error. Examples of errors are memory violation or null pointer.
Result: USFN process is terminated, and ordinary processing continues.

3. USFN self-terminates. For example, the USFN may detect a fatal internal error, or it may halt if it detects a certain condition.
Result: USFN process is terminated, and ordinary processing continues.

4. USFN fails to complete in time.
Result: Solartron 1051 software attempts to coordinate data processing with the USFN, and posts warning messages.

Miscellaneous Observations

* Development of the user function can take place in simulation mode, in parallel with real -time data acquisition.

* Development must be accomplished on a Digital Alpha workstation, since Solartron Ltd. does not support Sun computer products.

* Solartron Ltd. uses Free Software Foundation products for development, such as gcc, gnumake, and gdb. For example, Digital's version of the make utility does not work with Solartron code.

* Configuration files of the 1051 software may have to be modified to accommodate the user function, such as user_ch and chan_machine.

* Formal Solartron documentation for the user function is limited to Appendix A in the System Manager Guide, including a source code listing of a sample user function. One of the chief difficulties in preparing new code is locating data elements.

* Ames Research Center is the only Solartron Ltd. customer that utilizes the user function.

Accomplishments

1. Example user function as supplied in the documentation.

2. Simple modified user function.
Number of user function calls, user channel number, and smax are displayed in a monitor window and stored in an ASCII file.

3. Complex modified user function.
Data were intercepted from three user data channels and diverted into three derived data channels. Channel data were written into a file and/or displayed in a client window on a remote monitor depending upon the channel configurations. The output file name and path were parameterized. Output data included the function call count, channel number, DC_LEVEL, SYNC_PK_TO_PK, HARMONIC_RMS, and SYNC_RMS_TOTAL. Data were output only if actually present for any given channel as determined by a test condition for their presence.

4. User function modified to transfer data to RTWorks.

Sample Output Data

Solartron Simulation Mode
First Three Integral Harmonics

```
Call count = 1 user_chan_nr = 13000 centiseconds = 630345500
harmonic 1 phase 0.136638 magnitude 4.349309
harmonic 2 phase 0.040991 magnitude 1.304793
harmonic 3 phase 0.068319 magnitude 2.174654
Call count = 2 user_chan_nr = 13000 centiseconds = 630345600
harmonic 1 phase 0.161024 magnitude 5.125537
harmonic 2 phase 0.048307 magnitude 1.537661
harmonic 3 phase 0.080512 magnitude 2.562769
Call count = 3 user_chan_nr = 13000 centiseconds = 630345700
harmonic 1 phase 0.084241 magnitude 2.681468
harmonic 2 phase 0.025272 magnitude 0.804440
harmonic 3 phase 0.042120 magnitude 1.340734
Call count = 4 user_chan_nr = 13000 centiseconds = 630345800
harmonic 1 phase 0.103393 magnitude 3.291115
harmonic 2 phase 0.031018 magnitude 0.987334
harmonic 3 phase 0.051697 magnitude 1.645557
```

Sample Output Data

Solartron Simulation Mode

First 16 Integral Harmonics & 5 Non-Integral Harmonics

```
Call count = 1 user_chan_nr = 13000 centiseconds = 672923300
harmonic 1 phase 0.136638 magnitude 4.349309
harmonic 2 phase 0.040991 magnitude 1.304793
harmonic 3 phase 0.068319 magnitude 2.174654
harmonic 4 phase 1.570796 magnitude 0.000000
harmonic 5 phase 0.001571 magnitude 0.050000
harmonic 6 phase 0.001571 magnitude 0.050000
harmonic 7 phase 0.001571 magnitude 0.050000
harmonic 8 phase 0.001571 magnitude 0.050000
harmonic 9 phase 0.001571 magnitude 0.050000
harmonic 10 phase 0.001571 magnitude 0.050000
harmonic 11 phase 0.001571 magnitude 0.050000
harmonic 12 phase 0.001571 magnitude 0.050000
harmonic 13 phase 0.001571 magnitude 0.050000
harmonic 14 phase 0.001571 magnitude 0.050000
harmonic 15 phase 0.001571 magnitude 0.050000
harmonic 16 phase 0.001571 magnitude 0.050000
band 1 rms_in_band 0.430877 max_value_in_band 0.043088 max_posn_in_band 32.000000
band 2 rms_in_band 0.430877 max_value_in_band 0.043088 max_posn_in_band 32.000000
band 3 rms_in_band 0.430877 max_value_in_band 0.043088 max_posn_in_band 32.000000
band 4 rms_in_band 0.430877 max_value_in_band 0.043088 max_posn_in_band 64.000000
band 5 rms_in_band 0.430877 max_value_in_band 0.043088 max_posn_in_band 96.000000
```

Future Ideas

1. Test data transfer from the Solartron user function into RTWorks via an ASCII file.

2. Consider data transfer from Solartron to RTWorks via Unix sockets.

3. Test calling TEAMS-RT via the Solartron user function.